

Amendments to the Claims

1 Claim 1 (currently amended): A computer-implemented method of selecting an abstraction level
2 to use when generating parser output, comprising ~~a step of~~ steps of:

3 requesting, by an application program, generation of parser output[[,]] by a parser that
4 parses an input[[,]]; and

5 receiving, by the application program from the parser, output generated by the parser from
6 the input, wherein ~~such that~~ the generated output adheres to a different syntax level than a syntax
7 level used when validating the input.

1 Claim 2 (currently amended): The method according to Claim 1, wherein the ~~validation~~ validating
2 of the input is performed by the parser.

1 Claim 3 (original): The method according to Claim 1, wherein the input is a structured document.

1 Claim 4 (currently amended): The method according to Claim [[4]] 3, wherein the structured
2 document is encoded in Extensible Markup Language (“XML”).

1 Claim 5 (currently amended): The method according to Claim 1, wherein the generated output
2 comprises ~~one or more~~ at least one object ~~representations~~ representation generated from the
3 input.

1 Claim 6 (currently amended): The method according to Claim 1, wherein the parser is a

2 validating parser that also performs the ~~validation~~ validating of the input.

1 Claim 7 (currently amended): The method according to Claim 1, wherein the requesting step
2 further comprises the step of specifying a schema name of a schema to which the generated
3 output must adhere.

1 Claim 8 (currently amended): The method according to Claim 1, wherein the requesting step
2 further comprises the step of specifying a schema name of a schema to be used by the parser when
3 generating the output.

1 Claim 9 (original): The method according to Claim 8, wherein the schema name is specified as a
2 feature of the parser.

1 Claim 10 (currently amended): The method according to Claim 8, wherein the schema name is
2 specified by ~~[[an]]~~ the application program ~~for which an instance of the parser is created.~~

1 Claim 11 (currently amended): The method according to Claim 1, wherein the syntax level used
2 for the validating of the input is specified in the input.

1 Claim 12 (original): The method according to Claim 11, wherein the specification in the input
2 uses a schema location construct in the input.

1 Claim 13 (currently amended): A computer-implemented method of casting objects, further
2 comprising ~~a step of~~ steps of:

3 validating an input according to a first syntax level while generating output objects, from
4 the input, according to a second syntax level; and
5 providing the generated output objects for use by an application program.

1 Claim 14 (original): The method according to Claim 13, wherein the second syntax level is a less-
2 restrictive version of the first syntax level.

1 Claim 15 (original): The method according to Claim 13, wherein the first syntax level is a more-
2 restrictive definition of the second syntax level.

1 Claim 16 (original): The method according to Claim 13, wherein the first syntax level is an
2 extension of the second syntax level.

1 Claim 17 (currently amended): The method according to Claim 13, wherein the first syntax level
2 represents ~~at least one~~ an extension of the second syntax level.

1 Claim 18 (original): The method according to Claim 13, wherein the first syntax level and the
2 second syntax level are defined using schemas.

1 Claim 19 (original): The method according to Claim 18, wherein the schema that defines the first

2 syntax level is an extension of the schema that defines the second syntax level.

1 Claim 20 (currently amended): The method according to Claim 13, wherein the first syntax level
2 represents ~~at least one extension of~~ a plurality of extensions to the second syntax level.

1 Claim 21 (currently amended): The method according to Claim 13, wherein the generated output
2 ~~adheres~~ objects adhere to the second syntax level.

1 Claim 22 (original): The method according to Claim 13, wherein the input adheres to an extended
2 schema that defines the first syntax level.

1 Claim 23 (currently amended): The method according to Claim 22, wherein the generated output
2 objects adhere ~~adheres~~ to a base schema that is extended by the extended schema.

1 Claim 24 (currently amended): A system for applying abstraction to object markup definitions,
2 further comprising:

3 a validating parser usable by a computer;

4 first means for using the validating parser to validate an input document expressed as an
5 object markup definition, wherein the validation is performed according to a syntax level which
6 allows the object markup definition to be successfully validated; and

7 second means for using the validating parser to apply abstraction to the object markup
8 definition when generating [[an]] at least one output object for use by a computer application

9 therefrom, responsive to the first means, wherein the ~~application~~ applying of the abstraction
10 generates the at least one output object ~~object~~ according to a different syntax level which would
11 not allow the object markup definition to be successfully validated.

1 Claim 25 (currently amended): The system according to Claim 24, wherein the different syntax
2 level is requested by an application program that will consume the at least one generated output
3 object.

1 Claim 26 (currently amended): A computer program product for ~~improved~~ parsing of input, the
2 computer program product embodied on one or more computer-usable media and comprising:
3 computer-readable program code ~~[[means]]~~ for validating an input according to a first
4 schema, wherein the first schema defines a first syntax level that enables content in the input to be
5 successfully validated; and
6 computer-readable program code ~~[[means]]~~ for generating one or more output objects
7 according to a second schema, upon parsing the successfully-validated content in the input,
8 wherein the second schema defines a second syntax level that does not enable the content in the
9 input to be successfully validated.

1 Claim 27 (original): The computer program product according to Claim 26, wherein the first
2 syntax level is a more-restrictive version of the second syntax level.

1 Claim 28 (original): The computer program product according to Claim 26, wherein the first

2 schema is defined as an extension of the second schema.

1 Claim 29 (original): The computer program product according to Claim 26, wherein the first
2 schema is defined as an extension of some intermediate schema that extends the second schema.

1 Claim 30 (original): The computer program product according to Claim 26, wherein the second
2 schema is a base schema upon which one or more extensions are based, and wherein the second
3 schema is one of the extensions and is based either directly on the base schema or on an
4 intermediate schema that extends the base schema.

1 Claim 31 (currently amended): A computer-implemented method of ~~doing business by~~ providing
2 ~~improved~~ validation and parsing for clients, comprising steps of:

3 providing a validating parser that enables a client to dynamically select an abstraction level
4 for use when generating output from the validating parser;

5 obtaining an input document to be validated and parsed for the client;

6 validating the input document with the provided validating parser, wherein the validation
7 is performed according to a first syntax level associated with syntax specified in the input
8 document; and

9 generating output from the input document with the provided validating parser, for use by
10 the client, wherein the generated output has syntax that conforms to the abstraction level that has
11 been dynamically selected by the client and wherein the abstraction level is a refinement of the first
12 syntax level; ~~and~~

13 ——— charging a fee for at least one of the providing, obtaining, validating, and generating steps.